

What is claimed is:

1           1. A method for managing a cache, the method comprising:

2                   assigning a cache priority to each of a plurality of accessed item as a function of  
3                   the item's size, retrieval cost and access frequency;  
4                   dynamically updating cache priorities as items are accessed; and  
5                   determining which items to store in the cache as a function of cache priority.

1           2. The method of claim 1 further comprising:

2                   calculating an item's size relative to the size of the cache.

1           3. The method of claim 2 wherein calculating an item's size relative to the size of the  
2   cache further comprises:

3                   dividing the size of the item by the size of the cache.

1           4. The method of claim 1 further comprising:

2                   calculating an item's retrieval cost as a function of the item's retrieval time and the  
3                   item's size.

1           5. The method of claim 4 wherein calculating an item's retrieval cost as a function of the  
2   item's retrieval time and the item's size:

3                   dividing the retrieval time of the item by the size of the item.

1           6. The method of claim 1 further comprising:

2                   calculating an item's access frequency relative to access frequency for other items.

1           7. The method of claim 6 wherein calculating an item's access frequency relative to  
2 access frequency for other items further comprises:

3           dividing a number of requests for the item during a period of time by a total  
4           number of requests for items during the period of time.

1           8. The method of claim 1 further comprising:

2           calculating cache priority for an item by multiplying the item's size, retrieval cost  
3           and access frequency.

1           9. The method of claim 1 further comprising:

2           each time access to an item is requested, determining whether the requested item  
3           has been assigned a cache priority; and  
4           performing a step from a group of steps consisting of:

5           responsive to determining that the requested item has not been assigned a  
6           cache priority, calculating a cache priority and assigning the  
7           calculated cache priority to the requested item; and  
8           responsive to determining that the requested item has been assigned a  
9           cache priority, updating the cache priority to reflect the request for  
10          the item.

1           10. The method of claim 1 further comprising:

2           maintaining a sorted list of associations between each accessed item and its cache  
3           priority;

4 each time access to an item is requested, determining whether the requested item

5 has been assigned a cache priority by reading the sorted list; and

6 performing a step from a group of steps consisting of:

7 responsive to determining that the requested item has not been assigned a

8 cache priority, calculating a cache priority and adding an entry

9 associating the requested item with the cache priority to the sorted

10 list; and

11 responsive to determining that the requested item has been assigned a

12 cache priority, updating the requested item's entry in the sorted list

13 to reflect the request for the item.

1 11. The method of claim 1 wherein determining which items to store in the cache as a  
2 function of cache priority further comprises:

3 receiving a request for an item not in the cache;

4 retrieving the item;

5 determining that the cache is full;

6 comparing the cache priority of the retrieved item to the cache priority of each

7 item in the cache; and

8 performing a step from a group of steps consisting of:

9 responsive to determining that the cache priority of at least one item in the

10 cache is lower than the cache priority of the retrieved item,

11 overwriting a cached item with the lowest cache priority with the

12 retrieved item; and

13 responsive to determining that no item in the cache has a cache priority  
14 lower than the retrieved item, not storing the retrieved item in the  
15 cache.

1 12. A computer readable medium containing a computer program product for managing  
2 a cache, the computer readable medium comprising:  
3 program code for assigning a cache priority to each of a plurality of accessed item  
4 as a function of the item's size, retrieval cost and access frequency;  
5 program code for dynamically updating cache priorities as items are accessed; and  
6 program code for determining which items to store in the cache as a function of  
7 cache priority.

1 13. The computer program product of claim 12 further comprising:  
2 program code for calculating an item's size relative to the size of the cache by  
3 dividing the size of the item by the size of the cache.

1 14. The computer program product of claim 12 further comprising:  
2 program code for calculating an item's retrieval cost as a function of the item's  
3 retrieval time and the item's size by dividing the retrieval time of the item  
4 by the size of the item.

1 15. The computer program product of claim 12 further comprising:  
2 program code for calculating an item's access frequency relative to access  
3 frequency for other items by dividing a number of requests for the item

4 during a period of time by a total number of requests for items during the  
5 period of time.

1 16. The computer program product of claim 12 further comprising:

2 program code for calculating cache priority for an item by multiplying the item's  
3 size, retrieval cost and access frequency.

1 17. The computer program product of claim 12 further comprising:

2 program code for, each time access to an item is requested, determining whether  
3 the requested item has been assigned a cache priority; and  
4 program code for performing a step from a group of steps consisting of:

5 responsive to determining that the requested item has not been assigned a  
6 cache priority, calculating a cache priority and assigning the  
7 calculated cache priority to the requested item; and  
8 responsive to determining that the requested item has been assigned a  
9 cache priority, updating the cache priority to reflect the request for  
10 the item.

1 18. The computer program product of claim 12 further comprising:

2 program code for maintaining a sorted list of associations between each accessed  
3 item and its cache priority;

4 program code for, each time access to an item is requested, determining whether  
5 the requested item has been assigned a cache priority by reading the sorted  
6 list; and

7 program code for performing a step from a group of steps consisting of:

8 responsive to determining that the requested item has not been assigned a  
9 cache priority, calculating a cache priority and adding an entry  
10 associating the requested item with the cache priority to the sorted  
11 list; and  
12 responsive to determining that the requested item has been assigned a  
13 cache priority, updating the requested item's entry in the sorted list  
14 to reflect the request for the item.

1 19. The computer program product of claim 12 wherein the program code for  
2 determining which items to store in the cache as a function of cache priority further comprises:  
3 program code for receiving a request for an item not in the cache;  
4 program code for retrieving the item;  
5 program code for determining that the cache is full;  
6 program code for comparing the cache priority of the retrieved item to the cache  
7 priority of each item in the cache; and  
8 program code for performing a step from a group of steps consisting of:  
9 responsive to determining that the cache priority of at least one item in the  
10 cache is lower than the cache priority of the retrieved item,  
11 overwriting a cached item with the lowest cache priority with the  
12 retrieved item; and  
13 responsive to determining that no item in the cache has a cache priority  
14 lower than the retrieved item, not storing the retrieved item in the  
15 cache.

1        20. A computer system for managing a cache, the computer system comprising:  
2                means for assigning a cache priority to each of a plurality of accessed item as a  
3                function of the item's size, retrieval cost and access frequency;  
4                means for dynamically updating cache priorities as items are accessed; and  
5                means for determining which items to store in the cache as a function of cache  
6                priority.

1        21. The computer system of claim 20 further comprising:  
2                means for calculating an item's size relative to the size of the cache by dividing  
3                the size of the item by the size of the cache.

1        22. The computer system of claim 20 further comprising:  
2                means for calculating an item's retrieval cost as a function of the item's retrieval  
3                time and the item's size by dividing the retrieval time of the item by the  
4                size of the item.

1        23. The computer system of claim 20 further comprising:  
2                means for calculating an item's access frequency relative to access frequency for  
3                other items by dividing a number of requests for the item during a period  
4                of time by a total number of requests for items during the period of time.

1        24. The computer system of claim 20 further comprising:  
2                means for calculating cache priority for an item by multiplying the item's size,  
3                retrieval cost and access frequency.

1        25. The computer system of claim 20 further comprising:

2                means for, each time access to an item is requested, determining whether the

3                requested item has been assigned a cache priority; and

4                means for performing a step from a group of steps consisting of:

5                responsive to determining that the requested item has not been assigned a

6                cache priority, calculating a cache priority and assigning the

7                calculated cache priority to the requested item; and

8                responsive to determining that the requested item has been assigned a

9                cache priority, updating the cache priority to reflect the request for

10              the item.

1        26. The computer system of claim 20 further comprising:

2                means for maintaining a sorted list of associations between each accessed item

3                and its cache priority;

4                means for, each time access to an item is requested, determining whether the

5                requested item has been assigned a cache priority by reading the sorted

6                list; and

7                means for performing a step from a group of steps consisting of:

8                responsive to determining that the requested item has not been assigned a

9                cache priority, calculating a cache priority and adding an entry

10              associating the requested item with the cache priority to the sorted

11              list; and



12 responsive to determining that the requested item has been assigned a  
13 cache priority, updating the requested item's entry in the sorted list  
14 to reflect the request for the item.

1 27. The computer system of claim 20 wherein the means for determining which items to  
2 store in the cache as a function of cache priority further comprises:

3 means for receiving a request for an item not in the cache;  
4 means for retrieving the item;  
5 means for determining that the cache is full;  
6 means for comparing the cache priority of the retrieved item to the cache priority  
7 of each item in the cache; and  
8 means for performing a step from a group of steps consisting of:  
9 responsive to determining that the cache priority of at least one item in the  
10 cache is lower than the cache priority of the retrieved item,  
11 overwriting a cached item with the lowest cache priority with the  
12 retrieved item; and  
13 responsive to determining that no item in the cache has a cache priority  
14 lower than the retrieved item, not storing the retrieved item in the  
15 cache.

1 28. A computer system for managing a cache, the computer system comprising:  
2 a software portion configured to assign a cache priority to each of a plurality of  
3 accessed item as a function of the item's size, retrieval cost and access  
4 frequency;

5 a software portion configured to dynamically update cache priorities as items are  
6 accessed; and  
7 a software portion configured to determine which items to store in the cache as a  
8 function of cache priority.

1 29. The computer system of claim 28 further comprising:

2 a software portion configured to calculate an item's size relative to the size of the  
3 cache by dividing the size of the item by the size of the cache.

1 30. The computer system of claim 28 further comprising:

2 a software portion configured to calculate an item's retrieval cost as a function of  
3 the item's retrieval time and the item's size by dividing the retrieval time of  
4 the item by the size of the item.

1 31. The computer system of claim 28 further comprising:

2 a software portion configured to calculate an item's access frequency relative to  
3 access frequency for other items by dividing a number of requests for the  
4 item during a period of time by a total number of requests for items during  
5 the period of time.

1 32. The computer system of claim 28 further comprising:

2 a software portion configured to calculate cache priority for an item by  
3 multiplying the item's size, retrieval cost and access frequency.

1 33. The computer system of claim 28 further comprising:

2 a software portion configured to determine, each time access to an item is  
3 requested, whether the requested item has been assigned a cache priority;  
4 and  
5 a software portion configured to perform a step from a group of steps consisting  
6 of:  
7 responsive to determining that the requested item has not been assigned a  
8 cache priority, calculating a cache priority and assigning the  
9 calculated cache priority to the requested item; and  
10 responsive to determining that the requested item has been assigned a  
11 cache priority, updating the cache priority to reflect the request for  
12 the item.

1 34. The computer system of claim 28 further comprising:

2 a software portion configured to maintain a sorted list of associations between  
3 each accessed item and its cache priority;  
4 a software portion configured to determine, each time access to an item is  
5 requested, whether the requested item has been assigned a cache priority  
6 by reading the sorted list; and  
7 a software portion configured to perform a step from a group of steps consisting  
8 of:  
9 responsive to determining that the requested item has not been assigned a  
10 cache priority, calculating a cache priority and adding an entry  
11 associating the requested item with the cache priority to the sorted  
12 list; and

13 responsive to determining that the requested item has been assigned a  
14 cache priority, updating the requested item's entry in the sorted list  
15 to reflect the request for the item.

1 35. The computer system of claim 28 wherein the software portion configured to  
2 determine which items to store in the cache as a function of cache priority further comprises:  
3 a software portion configured to receive a request for an item not in the cache;  
4 a software portion configured to retrieve the item;  
5 a software portion configured to determine that the cache is full;  
6 a software portion configured to compare the cache priority of the retrieved item  
7 to the cache priority of each item in the cache; and  
8 a software portion configured to perform a step from a group of steps consisting  
9 of:  
10 responsive to determining that the cache priority of at least one item in the  
11 cache is lower than the cache priority of the retrieved item,  
12 overwriting a cached item with the lowest cache priority with the  
13 retrieved item; and  
14 responsive to determining that no item in the cache has a cache priority  
15 lower than the retrieved item, not storing the retrieved item in the  
16 cache.